



(12) **EUROPEAN PATENT APPLICATION**

(43) Date of publication:
16.02.2011 Bulletin 2011/07

(51) Int Cl.:
G06F 21/00 (2006.01)

(21) Application number: **10009390.5**

(22) Date of filing: **26.04.2004**

(84) Designated Contracting States:
DE FR GB

(62) Document number(s) of the earlier application(s) in accordance with Art. 76 EPC:
04009851.9 / 1 591 861

(71) Applicant: **SAP AG**
69190 Walldorf (DE)

(72) Inventors:
• **Steinmann, Jörg**
69190 Walldorf (DE)
• **Karthikeyan, A.**
69190 Walldorf (DE)
• **Kacker Himanshu**
69190 Walldorf (DE)

• **Marar Mohan**
69190 Walldorf (DE)
• **Boruah Jayanta**
69190 Walldorf (DE)
• **Gentes Wolfgang**
69190 Walldorf (DE)

(74) Representative: **Müller-Boré & Partner**
Patentanwälte
Grafinger Straße 2
81671 München (DE)

Remarks:

This application was filed on 09-09-2010 as a divisional application to the application mentioned under INID code 62.

(54) **Method, computer program and device for executing actions using data sets**

(57) According to the invention there is provided a method for executing actions using data sets, said data sets having a specified property, said specified property being indicated by a mark associated with each data set and being associated with a specified maximum number of actions to be executed, the method comprising the steps of:

- reading an action counter associated with a data set, said action counter indicating the number of executed actions using said data set;
- comparing (R18) said number of executed actions with said maximum number of actions;
- executing (R22) an action if said number of executed actions is lower than said maximum number of actions;
- increasing (R20) said action counter if an action is executed;
- separately storing all actions already performed with a particular data set in an action table or list;
- generating the value of the counter by summing up the number of already performed actions for a particular data set.

Furthermore, there is provided a computer program product and a device.

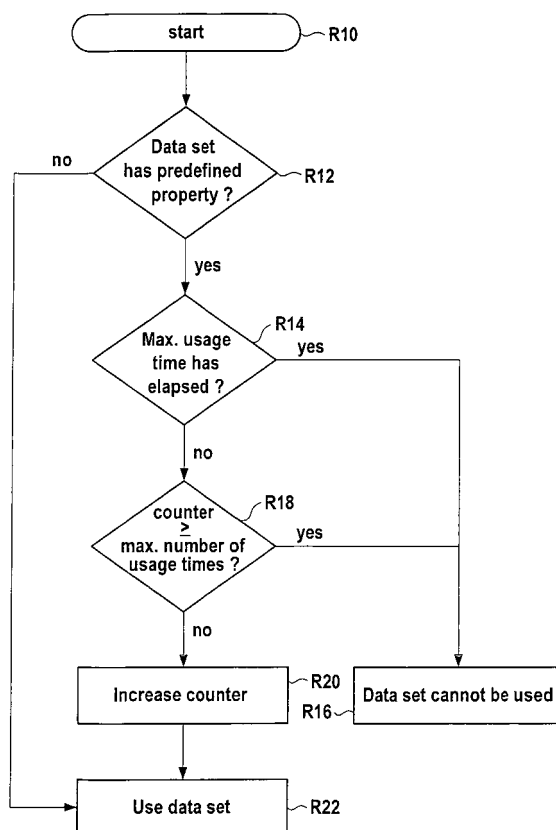


FIG. 5

Description

[0001] The present invention relates to a method, a computer program and a device for executing actions using data sets.

[0002] When executing actions on data sets, particular properties of the data sets need to be taken into account. The systems known in the prior art however only offer a limited functionality.

[0003] It is an object of the present invention to provide a method, a computer program and a device for executing actions using data sets providing an enhanced functionality.

[0004] This object is solved according to the present invention by a method having the features disclosed in claim 1, a computer program having the features disclosed in claim 5 and a device having the features disclosed in claim 6. Preferred embodiments are subject of the dependent claims.

[0005] According to the present invention there is provided a method for executing actions using data sets, said data sets having a specified (predetermined or predeterminable) property, said specified property being indicated by a mark associated with each data set and being associated with a specified (predetermined or predeterminable) maximum number of actions to be executed, the method comprising the steps of:

- reading an action counter associated with a data set, said action counter indicating the number of executed actions using said data set;
- comparing said number of executed actions with said maximum number of actions;
- executing an action if said number of executed actions is lower than said maximum number of actions;
- increasing said action counter if an action is executed.

[0006] Preferably the data sets have a specified (predetermined or predeterminable) maximum use period or maximum use date, said method further comprising the steps of:

- comparing the elapsed use period with the maximum use period before executing said action; and
- executing said action only if said elapsed use period is lower than, i.e. does not exceed said maximum use period.

[0007] The method preferably further comprises the step of decreasing said action counter if said action to be executed cannot be executed.

[0008] The method can further comprise the steps of:

- tracking a reaction in response to said executed action; and
- removing said mark if a specified (predetermined or predeterminable) reaction in response to the action is tracked.

[0009] According to the invention there is furthermore provided a computer program product for executing actions using data sets comprising program parts for carrying out a method according to one of the preceding claims.

[0010] Moreover, according to the invention there is provided a device for executing actions using data sets, said data sets having a specified (predetermined or predeterminable) property, said specified property being indicated by a mark associated with each data set and being associated with a specified (predetermined or predeterminable) maximum number of actions to be executed, said device comprising:

- an action counter associated with each data set, said action counter indicating the number of executed actions using said data set;
- a memory for storing said maximum number of actions;
- a comparing device for comparing the value of said action counter and said maximum number of actions of a respective data set, wherein said system is suitable for executing an action if said number of executed actions is lower than said maximum number of actions;
- increasing said action counter if an action is executed.

[0011] The computer device can furthermore comprise a tracking device for tracking a reaction in response to said executed action.

[0012] Further features and advantages will become apparent from the following detailed description of preferred embodiments of the present invention with reference to the drawings.

Fig. 1 schematically shows a system for importing an processing data.

Fig. 2 shows a schematic overview of a method of uploading and processing data in a preferred table structure.

Figs. 3A-3H show screen-shots of a user interface for operating the table structure.

Figs. 4A-4D show screen-shots of a user interface provided for a mapping function.

5 Fig. 5 shows a flowchart describing checks performed in data sets before they are used.

Fig. 6 shows a flowchart describing the deletion of data sets.

10 Fig. 7 shows a flowchart describing another alternative of the deletion of data sets

[0013] A user system 10 for importing and processing data is schematically shown in Fig. 1. The user system 10 comprises a first section 12 which can be used for importing and processing external data. The user system furthermore comprises at least one second section 14 in which data contained in the user system is processed.

15 **[0014]** A table structure and method for importing and processing large amounts of data sets into said user system 10 will be described in the following. As a preferred embodiment, the importing and processing of address data in a system will be described hereinafter.

[0015] External data comprising a plurality of data sets to be imported into a system are provided on an electronic storage medium, like a CD, or are received via ISDN, email, and so on. The data sets are provided in a predefined format in a data file e.g. ASCII, Excel, or dBase. The data sets comprise data segments of different categories. For address data such categories could be first name, last name, street, house number, city, etc. The segments of a data set are separated by a separator in the data file. The separator can be e.g. comma, semicolon, tab or any other character.

[0016] Fig. 2 shows a schematic overview of a method of uploading and processing data in a preferred table structure.

25 **[0017]** External data to be imported into the system are first uploaded without any further processing (step S10). The data sets uploaded in the user system are referred to as "data list". The uploaded data sets are mapped (step S20). In particular the data structure of the uploaded data sets is transferred to the data structure of the system (to be described later). After mapping the data sets several checks are performed on the data sets in order to verify if particular properties of the data sets are correct. In the present case, as address data sets are used, a postal check (step S30) and a duplicate check (step S40) are performed. In the postal check it is verified whether the address data provided is correct with respect to general properties, e.g. it is checked whether the postal code correctly corresponds to the city given in one data set.

30 **[0018]** During the duplicate check step, it is checked whether duplicate data sets, i.e. substantially identical data sets or data sets having a specified (predetermined or predeterminable) range of identical data fields are present. During this check it can be verified whether a newly imported data set is substantially identical to a data set already present in the system or whether two newly imported data sets are identical. The duplicate check can e.g. be performed by comparing the components of the one data set (last name, first name, street, ...) with the same components of another data set.

35 **[0019]** After having performed the various checks, the imported data sets can be further processed (step S50) as will be described later.

[0020] In the following, a table structure used for importing and processing data and/or data sets will hereinafter be referred to as a "staging area".

40 **[0021]** The staging area comprises a plurality of tables into which data segments of the imported data sets and/or additional information relating to the data list is written.

[0022] The external data stored on a data storage medium is read line by line (data set by data set) and stored in table CRMD_MKTLIST_C. During this step, the data is not processed in any way. This means that in CRMD_MKTLIST_C the data is stored in a form similar to the form of the provided data in the data file. The data segments separated by the separator are written into the table CRMD_MKTLIST_C substantially without an processing. Thus the data sets as a whole are transferred from the data storage medium to the table CRMD_MKTLIST_C without being split up into separate parts.

45 **[0023]** The data stored in CRMD_MKTLIST_C is then mapped. During the mapping step the data sets stored in CRMD_MKTLIST_C are each divided up, whereby one or more segments of each data set are stored e.g. in either of tables CRMD_MKTLIST_ADR, CRMD_MKTLIST_ORG, CRMD_MKTLIST_PER, CRMD_MKTLIST_CEN. In CRMD_MKTLIST_ADR data segments of each data set relating to address properties (e.g. street, postal code, city, ...) of the data sets are stored. In CRMD_MKTLIST_ORG data segments of each data set relating to the data set's organizational properties (e.g. company name, ...) are stored. In CRMD_MKTLIST_PER data segments of each data set relating to the person (e.g. first name, last name, age, sex, ...) are stored. In CRMD_MKTLIST_CEN data segments of each data set relating to containing additional information (e.g. contact person of a company, ...) are stored.

50 **[0024]** If the data sets relate to other types of data, other suitable tables can be used.

[0025] A table CRMD_MKTLIST_E is provided in which error messages for errors which occur during the processing of the data are logged.

[0026] Furthermore, there is provided a plurality of tables which form a so called "administrative area". In the present system the following tables are provided:

CRMD_MKTLIST_H
 CRMD_MKTLIST_T
 CRMD_MKTLIST_PH
 CRMD_MKTLIST_S
 CRMD_MKTLIST_L
 CRMD_MKTLIST_I

[0027] In CRMD_MKTLIST_H header information and attributes (e.g. origin of the data, type of the data, costs) relating to the list are stored. In CRMD_MKTLIST_T additional information (e.g. short description, language dependent) relating to the list is stored. In CRMD_MKTLIST_PH package header information and attributes is stored, e.g. the mapping format used or the name of the provided file containing the data sets to be imported. In CRMD_MKTLIST_S the steps which have already been processed for a specific data list are indicated. In CRMD_MKTLIST_L builds the connection between the header and the list itself. In particular a reference to CRMD_MKTLIST_H is stored in this table. In CRMD_MKTLIST_I a link or reference to a created data object (to be described later) is stored for each data set.

[0028] In the following, examples of the above mentioned tables are shown:

Header table CRMD_MKTLIST_H		
Field	Data type / -element	Meaning
CLIENT	MANDT	Client
LIST_GUID	RAW16	GUID of the list
LIST_ID	CHAR28	ID of the list
LIST_ID_LOGSYS	CHAR10	Logical system where the list was created
ORIGIN	CHAR6	Origin of the list
TYPE	CHAR2	Type of the list (bought, rented)
VALID_DATE_TO	DATS8	Recommended End-of-Use Date
MAX_USAGE_NO	NUMC4	Permitted No. of Uses for Address List
MAX_USAGE_DATE	DATS8	Permitted End-of-Use Date
NO_GROSS	NUMC8	Gross number
COSTS	CURR15	Costs
COSTS_CURRENCY	CUKY5	Currency of costs
CREATED_AT	DEC15	Creation date
CREATED_BY	CHAR12	Creation user
CHANGED_AT	DEC15	Change date
CHANGED_BY	CHAR12	Change user
DELETED	CHAR1	Flag which indicates that list is deleted
APPL_LOG_ID	CHAR22	ID of application log of deletion

Item table CRMD_MKTLIST_I		
Field	Data type / -element	Meaning
CLIENT	MANDT	Client
ITEM_GUID	RAW16	GUID of the line
LIST_GUID	RAW16	GUID of the list
BP_GUID_ORG	RAW16 (type	GUID of the assigned business partner organization)

(continued)

Item table CRMD_MKTLIST_I		
<i>Field</i>	<i>Data type / -element</i>	<i>Meaning</i>
BP_GUID_PER	RAW16 (type	GUID of the assigned business partner person)
POSTAL	CHAR1	Flag that the entry has postal errors
DUPLICATE	CHAR1	Flag that the entry is a duplicate

Text table		
<i>CRMD_MKTLIST_T Field</i>	<i>Data type / -element</i>	<i>Meaning</i>
CLIENT	MANDT	Client
LIST_GUID	RAW16	GUID of the list
LIST_LG	LANG1	Language
LIST_DESCR	CHAR60	Description

Header table CRMD_MKTLIST_PH		
<i>Field</i>	<i>Data type / -element</i>	<i>Meaning</i>
CLIENT	MANDT	Client
PACKAGE_GUID	RAW16	GUID of the package
SET_GUID	RAW16	GUID of the profile set
SET_DESCRIPTION	CHAR72	description of a profile set
EXISTING_SET	CHAR1	
PF_GUID	RAW16	GUID of a profile
TG_GUID	RAW16	GUID of the target group
TG_DESCRIPTION	CHAR72	description of a target group
MAPPING_FORMAT	CHAR6	format of an external list
SEPARATOR_TYPE	CHAR1	Field separator type
SEPARATOR_CHAR	CHAR1	Field separator character
FILE	CHAR255	Location of the file
FILE_STORAGE	CHAR1	Storage place of file: workstation or server

Package-List-Assignment table CRMD_MKTLIST_L		
<i>Field</i>	<i>Data type / -element</i>	<i>Meaning</i>
CLIENT	MANDT	Client
PACKAGE_GUID	RAW16	GUID of the package
LIST_GUID	RAW16	GUID of the list

Package-Workflow-Steps-Assignment table CRMD_MKTLIST_S		
Field	Data type / - element	Meaning
CLIENT	MANDT	Client
PACKAGE_GUID	RAW16	GUID of the package
STEP	CHAR1	Step (U, M, P, D, B, T, C, R)
FLAG	CHAR1	flag, which indicates that step is selected
STATUS	CHAR1	defines, what the process status of the step is (active, finished or error)
ATTR_ERROR	CHAR1	Determines, what the workflow has to do, if the step has an error (break or continue)
REPEAT_ALL	CHAR1	Determines: wants the user to repeat the step for all entries in the list or only for the failed, if in the step occurs an error
APPL_LOG_ID	CHAR22	Application Log: Handle of a protocol

[0029] A graphical user interface for operating the staging area will be described with reference to Figs. 3A to 3H.

[0030] The user interface comprises a field 30 for inputting an ID of a list to be created and a field 32 for inputting a description of the list. Furthermore, the user interface comprises a plurality of tab strips 34-42 for selecting specified input screens described hereinafter.

[0031] Fig. 3A shows a basic screen for inputting basic data relating to the data file to be uploaded.

[0032] If the user wants to upload a new data file he must create a new data list. The user enters an ID of the data list in field 30 (in the present case: PERSON_LIST) and a description of the list in field 32. On tab strip 32 the user enters or can enter attributes such as the origin in field 44, the type in field 46, the costs in field 48, the permitted number of uses in field 50 or permitted end of use date in field 52 (if the data sets can only be used for a predefined number of times and/or a predefined usage period, to be described later), recommended end of use date in field 54 and/or the gross number in field 56 (= number of expected data sets in the file). This information input in region 62 is stored in tables CRMD-MKTLIST_H and CRMD-MKTLIST_T.

[0033] Furthermore, the user enters the file name (with complete path) in field 58 and the field separator used in the data file to be imported (comma, semicolon, tab, other character) in field 60. A so called "package" is created and the information input in region 64 is stored in table CRMD_MKTLIST_PH.

[0034] Fig. 3B shows a screen for inputting information relating to process steps to be performed.

[0035] On tab strip 36 the user can mark in the fields 68-74 the steps or processing that should be processed within one background job, i.e. in one run. In the shown screen, the user can select one, several or all of 'Upload File' (field 66), 'Map Data' (field 68), 'Postal Check' (field 70), 'Duplicate Check' (field 72), and two steps for further processing (fields 74, 76) ('Business Partner Creation' and 'Target Group Creation').

[0036] If the step 'Map Data' (field 66) is marked a mapping format must be given (to be described later).

[0037] The user can specify in field 80 the start type, i.e. whether the processing is to start immediately or at a specified later time.

[0038] If the user presses the save button 78, the entered data is saved.

[0039] The input data is then written into tables CRMD_MKTLIST_PH and CRMD_MKTLIST_L by the system. Table CRMD_MKTLIST_S contains all steps for the package, wherein the status of every step is initial.

[0040] Fig. 3C shows a screen for inputting additional information relating to the list to be imported. On tab strip 38 there is provided a field 82 into which the user can enter a long text.

[0041] In the following several examples for processing are given.

[0042] The user marks the upload step (field 66), gives a start date/time (field 80) and saves the list (button 78). The file is read from application server and its content is stored in table CRMD_MKTLIST_C. In particular the data sets stored in the data file are read data set by data set and written into table CRMD_MKTLIST_C.

[0043] In table CRMD_MKTLIST_S the status of step U ('upload') is updated. If errors occur the step can be repeated. In that case only those data sets of the file are processed which were not transferred to CRMD_MKTLIST_C earlier. If the step was processed without any error it cannot be repeated again.

[0044] The user marks the mapping step (field 68), gives a start date/time (field 80) and saves the list (button 78). Once the input by the user is terminated and/or the start time is reached, the processing starts. The previously uploaded data is read from table CRMD_MKTLIST_C data set by data set, mapped and stored in the other tables of the staging area: CRMD_MKTLIST_ORG

CRMD_MKTLIST_PER
 CRMD_MKTLIST_ADR
 CRMD_MKTLIST_CEN

[0045] In particular, an iterative processing takes place in which respective data segments of each data set are read and subsequently written into one of the above tables.

[0046] In table CRMD_MKTLIST_S the status of step M ('mapping') is updated. In table CRMD_MKTLIST_I the items are inserted. If errors occur the step can be repeated for all data sets or only for the erroneous. At start of the repetition for all data sets the entries in CRMD_MKTLIST_E, CRMD_MKTLIST_ADR, CRMD_MKTLIST_ORG, CRMD_MKTLIST_PER and CRMD_MKTLIST_CEN are deleted for the repeated data sets. New entries in CRMD_MKTLIST_E, CRMD_MKTLIST_ADR, CRMD_MKTLIST_ORG, CRMD_MKTLIST_PER and CRMD_MKTLIST_CEN are created during the processing. The entries in CRMD_MKTLIST_I are updated only.

[0047] In case of repetition for erroneous data sets, only those data sets that are marked as erroneous (in table CRMD_MKTLIST_E) and those data sets that were not processed in previous execution are processed again. Entries in CRMD_MKTLIST_E are deleted just before creating new entries.

[0048] The user marks the step for postal check (field 68), gives a start date/time (field 80) and saves the list (button 78). Once the input by the user is terminated and/or the start time is reached, the processing starts. The previously mapped data is read from tables of the staging area

CRMD_MKTLIST_ORG
 CRMD_MKTLIST_PER
 CRMD_MKTLIST_ADR
 CRMD_MKTLIST_CEN

and checked against postal reference data. In table CRMD_MKTLIST_S the status of step P ('Postal Check') is updated. In table CRMD_MKTLIST_I the items are updated, field POSTAL is filled. This step is optional for the processing of the data.

[0049] If errors occur the step can be repeated for all data sets s or only for the erroneous. At start of the repetition the entries in CRMD_MKTLIST_E are deleted for the repeated data sets. New entries in CRMD_MKTLIST_E are created during the processing. The entries in CRMD_MKTLIST_I are updated only.

[0050] The user marks the step for duplicate check (field 72), gives a start date/time (field 80) and saves the list (button 78). Once the input by the user is terminated and/or the start time is reached, the processing starts. The previously mapped data is read from tables of the staging area

CRMD_MKTLIST_ORG
 CRMD_MKTLIST_PER
 CRMD_MKTLIST_ADR
 CRMD_MKTLIST_CEN

and checked if data sets of the lists exist already as business partner in the system.

[0051] In table CRMD_MKTLIST_S the status of step D ('Duplicate Check') is updated. In table CRMD_MKTLIST_I the items are updated, field DUPLICATE is filled. This step is optional for the processing of the data.

[0052] If errors occur the step can be repeated for all data sets or only for the erroneous. At start of the repetition the entries in CRMD_MKTLIST_E are deleted for the repeated data sets. New entries in CRMD_MKTLIST_E are created during the processing. The entries in CRMD_MKTLIST_I are updated only.

[0053] Fig. 3D shows a corresponding to the screen shown in Fig. 3B after the processing of the data has taken place in which tab strip 36 is visible.

[0054] After the marked steps are processed a traffic light 84-92 shows if errors occurred. By pressing the button associated with the traffic light 84-92 a popup window with detailed messages appears (see e.g. Fig. 3F).

[0055] The processing of the steps can be repeated, the user must mark the step again. For some steps the user can mark a second flag 94. If this flag 94 is marked all data sets are processed again, otherwise only the erroneous data sets are processed again.

[0056] A summary log with status information of all processed steps can be opened by pressing the log button 96 (see Figs. 3A and 3E).

[0057] Fig. 3E shows a summary log about status information of all processed steps and Fig. 3F shows a popup with detailed error information of a selected step.

[0058] Fig. 3G shows a screen with the tab strip 40. After the step 'Map Data' (to be described later in detail) is processed the result can be checked on this tab strip 40.

[0059] In Fig. 3H shows a screen with the tab strip 42. Here some key figures and administrative data are shown.

[0060] The processed data stored in the tables of the staging area can be use used in the user system for creating data objects for each data set. These data objects can then be further processed in the user system.

[0061] If a data object is created for a data set, a reference to the created data object is stored in the table CRMD_MKTLIST_I. The created data object is preferably stored centrally in the user system to be accessible for a plurality of

applications. Table CRMD_MKTLIST_I establishes a link between the data sets stored in the staging area and the centrally stored data object.

[0062] In the following a User Interface for mapping the read data sets stored in table CRMD_MKTLIST C will be described with reference to Figs. 4A to 4D. Figs. 4A to 4D show screen-shots of a graphical user interface provided for the mapping function.

[0063] Before a data list can be mapped a mapping format must be defined.

[0064] Fig. 4A shows a screen in which the user can edit some or more mapping features.

[0065] In the general section M10 there is provided a field 98 for inputting an ID of a mapping format to be used. A predefined mapping format previously stored can be used. Alternatively, a new mapping format can be created or an existing mapping format can be edited. Furthermore, in the field 100 a text description of the mapping format can be entered.

[0066] The file name of the imported file is entered in the field 102. Furthermore, the type of field separator is to be chosen in field 104.

[0067] Moreover, there are table controls M12 and M14 placed side to side. The table control M12 on the left (segment display section) has three fields 106, 108, 110 in which a position (column number of data segment in the imported file), a file preview (data segment of one data set in the file) and/or the target field name which indicates the target field to which the respective data segment is to be associated are displayed, respectively.

[0068] When creating a new mapping format this table control M12 is empty. When the data file for which the mapping format is to be used is selected, the file contents is read depending on the file separator selected. If a correct separator is selected the data is read and shown to the user in table control M12 with each row R1, R2, R3 in the table control M12 showing individual column values (data segments) of one data set of the data file in field 108. Since its only a preview, the first line of the data set is read and placed in the table control M12. If the user had chosen an invalid separator, the whole data set (first line) is shown in one single row (the first row R1 of the table control M12).

[0069] The table control M14 on the right (target display section) shows possible target fields defined in the staging area and their description. In the left table control M12 for every column (position) in the file a target field can be assigned in field 110.

[0070] A category column (field 112) appears in this control M14, which clearly demarcates the target fields which belong to an organization, organization address, person, person address and contact person function, in the case if address data are used. In case other types of data are used, other specifications can be given for the target fields. This table control displays only target fields.

[0071] Since now the user knows the file content for which the mapping has to be done, he can choose the target field by marking one of rows R4-R6 on the table control M14 and the source field by marking one of rows R1-R3 on the table control M12) and press the move button 114 which appears between these controls M12, M14.

[0072] Some target fields only allow predefined values defined in a customizing table. For those target fields a value mapping can be defined as will be described hereinafter. For the value assignment the tab 116 is provided.

[0073] As shown in Fig. 4A, for the data segment "PERS_TITLE_KEY", i.e. the title of the person, in the user system the values 0001 and 0002 are provided (column 122), 0001 representing "Mrs." and 0002 representing "Mr.". In the data sets provided, e.g. the title of a person is indicated by the usual expressions Mrs. and Mr.. Therefore, these usual expressions need to be substituted by the values used in the user system when importing the data sets into the user system.

[0074] Moreover, it is possible to assign a plurality of expressions used in the imported data sets to a specified (predetermined or predeterminable) value in the user system. A table control M16 is provided where the user can assign multiple values to the possible values of the mapped field. In the present example, the expressions "Mr.", "Mr", "Mister" and "MR" are all assigned to the value 0002 representing "Mr." in the user system.

[0075] Furthermore, there can be provided a field in which all possible values of a particular data segment of all data sets of a file for which value mapping is to be provided are displayed (not shown in Fig. 4A). Thus, the user can see all possible expressions used in the data file and assign these respectively to the values given in the user system.

[0076] Fig. 4B shows a screen corresponding to the screen of Fig. 4A, wherein the tab 118 is visible.

[0077] For those target fields that should have the same value for all data sets in the file a constant assignment can be defined.

[0078] The constant assignment can be used if all data sets of the data list should have the same value for a particular field. For example a data list has data sets only from the United States but the country information is not part of the file. The country field is assigned to the mapping format and the constant value US is assigned to this field. On tab strip 118 in field 124 the user can input the constant value to be assigned to all data sets.

[0079] Fig. 4C shows a screen corresponding to the screen of Fig. 4A, wherein the tab 120 is visible.

[0080] For those target fields that need a more complex mapping algorithm, a code can be programmed in field 126. In the shown user interface a so called "ABAP code" (Advanced Business Application Programming) can be defined.

[0081] The user interface does not allow the user to select this tab 120 unless and until the user selects a mapped

field from the table control M14, which holds the mapping fields. Now, the user can see the names of the input structure and the output structure (this is like input parameter and output parameter of a function module). The user can write executable routines in ABAP Code for this mapping field in this control. Care must be taken that the values that come in, are always in the input parameter and the output after the conversion should be assigned to the output parameter.

[0082] In the present example, the file contains no column with language information. Via coding the language information is derived from the country information.

[0083] Fig. 4D shows a screen corresponding to the screen of Fig. 4A, wherein the tab strip 122 is visible.

[0084] If global variables are needed for the ABAP coding they can be defined in tab strip 122 in field 128.

[0085] Data sets imported into the system can have a specified (predetermined or predeterminable) property. Such a property can be associated with a further constriction, e.g. that the data sets can only be used a specified (predetermined or predeterminable) number of times. In the present case, the specified property data can e.g. be that the data sets are rented from a data provider. Such rented data in general only have a limited number of uses and/or a limited time frame in which they can be used by the user. Therefore, the data have to be marked accordingly with a flag. In the present case, a so called "rented flag" is used. If a specified (predetermined or predeterminable) event occurs, e.g. a contacted person reacts to the an activity, the specified property of the data can be changed, i.e. the flag deleted.

[0086] In order to be able to check if accordingly marked data sets have already been used the maximum number of times, and can thus not be used again, or if the data set can still be used, a counter is associated with each data set. The counter indicates how many times the respective data set has already been used.

[0087] The checks to determine whether a data set can be used a further time have to be performed before the respective data set is used. However, such checks need only to be done if the data sets still has a specified property, i.e. is marked as rented in the present case.

[0088] In the following it will be described with reference to Fig. 5 how such checks are done.

[0089] In step R10 the processing of a specified data set starts. In step R12 it is checked whether the respective data set has a specified property. In case the data set does not have the specified property ("no" in step R12), the data set can be used without any further checks and the process proceeds to step R22. In case the data set has the specified property ("yes" in step R12), a check is made whether the maximum usage time for the data set has elapsed (step R14). If the maximum usage time for the data set has elapsed ("yes" in step R14), the data set cannot be used and the process proceeds to step R16. If the maximum usage time for the data set has not yet elapsed ("no" in step R14), a check is made whether the number of the counter associated with the data set is equal or larger than the maximum number of usage times of the data set (step R18). If the counter is equal or larger than the maximum number of usage times ("yes" in step R18), the data set cannot be used and the process proceeds to step R16. If the counter is less than the maximum number of usage times ("no" in step R18), i.e. the data set can still be used, the counter is increased (step R20) and the data set is used (step R22).

[0090] All actions already performed with a particular data set a separately stored in an action table or list so that each action can be tracked. Furthermore, in one preferred embodiment, the value of the counter is generated by summing up the number of already performed actions for a particular data set. By doing so, the data consistency can be enhanced as wrong values due to e.g. a system crash can be substantially avoided. In particular if e.g. an action is to take place, but the processing of the action is interrupted due to external factors.

[0091] If the action to be done with the data set cannot be executed properly, the data set can be used again, i.e. the counter must be decreased. This can e.g. be done by provided a check of the action table.

[0092] In case a specified reaction can be tracked in response to an action executed with a data set, the specified property can be removed from the respective data set. In such a case, the data set can be used freely substantially without the necessity of any of the above checks to be done.

[0093] The data set can be used in different sections of the user system. As already mentioned above, the user system can comprise a first section for importing and processing the data sets and at least one second section in which the data sets are used for interaction with still other parts of the system. If the data sets have the above defined specified property, there are restrictions given with respect to their use, as already partly discussed above. In order to maintain a consistency of the data, the counter is preferably stored in the first section and all sections using respective data sets inquire with the first section whether a data set can be used or not if the data set has the specified property.

[0094] In case the maximum usage time of a data list has run up and some or all of the data sets of this list still have the specified property, in many cases the data sets need to be removed from the system. However, as discussed above, it is possible that the data sets are also used in other sections of the system. As a consequence, the data not only has to be deleted in the first section into which the data has been imported but also in the other sections where the data sets are used.

[0095] In the following, the basic process of deleting data sets having a specified property in the first section of the user system will be described with reference to Fig. 6. For the following description it is assumed that only data sets are used and no data objects have been created.

[0096] The deletion of a data list is started in step D10. This process is in particular started if the maximum usage

time of a data list has expired.

[0097] In step D12 it is checked whether the data list has the specified property. In the present case this is a check whether the data list is rented or not. If the data list does not have the specified property ("no" in step D12), the data list does not have to be deleted (step D14). If the data list has the specified property ("yes" in step D12), step D16 is performed as long as the end of the list is not reached, i.e. as long as there are data sets in the data list.

[0098] In step D16 the following steps are performed preferably for each data set of the data list. The data set is read from the data list (step D18). It is checked, whether the read data set has the specified property (step D20). If the data set does not have the specified property ("no" in step D20), the data set does not have to be deleted (step D22). If data set has the specified property ("yes" in step D20), the data set is deleted in step D24. Furthermore, in this step an indicator or instruction set is sent to preferably all other sections of the system in which the data set is used. In the present case, the data sets are also used in the second section. Therefore an indicator is sent to the second section.

[0099] In the second section the indicator from the first section is received and the respective data set is anonymized, in particular the contents of the data set is filled with blank fields.

[0100] After having finished the processing described above, all data sets which had a specified property are removed from the system.

[0101] In the following, there will be given a description of a deletion process for the case in which data objects have been created from the data sets. In this case, all data from the staging area needs to be deleted. Furthermore, the created data objects having a specified property need to be deleted. A flow chart for such a deletion process is shown in Fig. 7.

[0102] In step D100 the deletion of the data list is started and in step D102 all data sets are deleted from the staging area, i.e. the tables CRMD_MKTLIST_E, CRMD_MKTLIST_C, CRMD_MKTLIST_ORG, CRMD_MKTLIST_PER, CRMD_MKTLIST_ADR, CRMD_MKTLIST_CEN.

[0103] The data stored in CRMD_MKTLIST_H is read (step D104), i.e. the ID of the list to be deleted is read. In step D106 it is checked whether the data list has the specified property. In case the data list does not have the specified property ("no" in step D106), the process proceeds to step D118. In case the data list has the specified property ("yes" in step D106), the reference to the data objects is read from table CRMD_MKTLIST_I (step D108).

[0104] For every data object associated with the list to be deleted (step D110), it is determined whether the data object has the specified property (step D112). In case the data object does not have the specified property ("no" in step D112), the process proceeds to step D114 and the data object is not deleted. In case the data list has the specified property ("yes" in step D112), the data object is deleted. Furthermore, in this step an indicator is sent to preferably all other sections of the system in which the data set is used. In the present case, the data sets are also used in the second section. Therefore an indicator is sent to the second section. In the second section a similar processing to the one described with reference to Fig. 6 is done. Alternatively, a blank data set can be sent to the second section to replace the data set associated with the deleted data set.

[0105] In step D118 the data in the tables CRMD_MKTLIST_S, CRMD_MKTLIST_L, CRMD_MKTLIST_PH are deleted. Then, the data in the table CRMD_MKTLIST_I is deleted (step D120). In the table CRMD_MKTLIST_H the delete flag is set (step D122).

[0106] After the deletion only the table CRMD_MKTLIST_H and CRMD_MKTLIST_T contain information about the deleted list.

[0107] The above description has been given for the importing and processing of address data set into a user system. However, other types of data can also be processed by the described structures. Such data could e.g. be measurement data which are to be evaluated in an evaluation system.

List of Reference Numerals

[0108]

10 user system

12 first section

14 second section

30 field

32 field

34-42 tab strip

44-60	field
62, 64	region
5 66-76	field
78	save button
80,82	field
10 84-92	traffic light
94	flag
15 96	log button
98-112	field
114	move button
20 116-122	tab strip
124-128	field
25 M10-M16	table control
R1-R6	row

30 Claims

1. A method for executing actions using data sets, said data sets having a specified property, said specified property being indicated by a mark associated with each data set and being associated with a specified maximum number of actions to be executed,
35 **characterized in that**
the method comprises the steps of:
 - reading an action counter associated with a data set, said action counter indicating the number of executed actions using said data set;
 - 40 - comparing (R18) said number of executed actions with said maximum number of actions;
 - executing (R22) an action if said number of executed actions is lower than said maximum number of actions;
 - increasing (R20) said action counter if an action is executed;
 - separately storing all actions already performed with a particular data set in an action table or list;
 - 45 - generating the value of the counter by summing up the number of already performed actions for a particular data set.
2. The method according to claim 1, wherein the data sets have a specified maximum use period, said method further comprising the steps of:
 - 50 - comparing (R14) the elapsed use period with the maximum use period before executing said action; and
 - executing (R22) said action only if said elapsed use period is lower than said maximum use period.
3. The method according to one of the preceding claims, further comprising the step of decreasing said action counter if said action to be executed cannot be executed.
- 55 4. The method according to one of the preceding claims, further comprising the steps of:
 - tracking a reaction in response to said executed action; and

- removing said mark if a specified reaction in response to the action is tracked.

5. The method according to any of the preceding claims, wherein the method comprises the further steps of:

importing external data comprising a plurality of data sets having the specified property into a user system (10), storing the external data in a first table; and preferably mapping the data stored in the first table, the data sets comprising data segments of different categories; and during the mapping step, dividing up the data sets stored in the first table, whereby one or more segments of each data set are stored in either of a plurality of second tables.

6. The method according to claim 5, further comprising the step of performing several checks on the data sets after mapping the data sets in order to verify if particular properties of the data sets are correct.

7. The method according to claim 6, wherein the checks include a postal check, the postal check including verifying whether address data provided is correct with respect to general properties, or checking whether a postal code correctly corresponds to a city given in one data set, and/or a duplicate check including checking whether duplicate data sets or data sets having a specified range of identical data fields are present, or verifying whether a newly imported data set is substantially identical to a data set already present in the system or whether newly imported data sets are identical.

8. The method according to any of the preceding claims 5 to 7, wherein one of the second tables has stored header information and attributes relating to a data list, a data list being the data sets uploaded in the user system (10); and/or wherein one of the second tables has stored additional information relating to the data list; and/or wherein one of the second tables has stored package header information and attributes; and/or wherein one of the second tables indicates the steps which have already been processed for a specific data list; and/or wherein one of the second tables builds a connection between the header and the data list itself.

9. The method according to any of the preceding claims 5 to 8, the method comprising the further steps of:

creating data objects for each data set, wherein one of the second tables has stored a link or reference to a created data object for each data set; starting deletion of a data list, and deleting all data sets in the first and second tables; reading in one of the second tables an ID of the data list to be deleted; checking whether the data list has a specified property; in case the data list has the specified property, reading from one of the second tables a reference to the data objects; for every data object associated with the list to be deleted, determining whether the data object has the specified property; not deleting a data object in case the data object does not have the specified property, and deleting a data object in case the data object has the specified property.

10. The method according to claim 9, the method further comprising deleting the data in a second table which indicates the steps which have already been processed for a specific data list; and/or in a second table which builds the connection between a header and the data list itself; and/or in a second table which stores package header information and attributes; and/or in a second table which stores a reference to a created data object; and/or wherein the method further comprises setting a delete flag in a second table which stores header information and attributes relating to the data list.

11. The method according to any of the preceding claims 5 to 10, wherein the method comprises the further steps of:

storing the action counter in a first section (12) of the user system (10), the first section (12) being for importing and processing the data sets, wherein the user system (10) has at least one second section (14) in which the data sets are used for interaction with still other parts of the system (10); inquiring, by all sections using respective data sets, with the first section (12) whether a data set can be used or not if the data set has the specified property.

12. The method according to any of the preceding claims 5 to 11, wherein the method comprises the further steps of:

deleting a data set having a specified property and sending an indicator or instruction set to all other sections of the user system (10) in which the data set is used;
receiving in a second section (14) the indicator from the first section (12);
anonymizing the respective data set in the second section (14).

5
13. A computer program product for executing actions using data sets comprising program parts for carrying out a method according to one of the preceding claims.

10
14. A computer device (10) for executing actions using data sets, said data sets having a specified property, said specified property being indicated by a mark associated with each data set and being associated with a specified maximum number of actions to be executed,
characterized in that
said device (10) comprises:

- 15
- an action counter associated with each data set, said action counter indicating the number of executed actions using said data set;
- a memory for storing said maximum number of actions;
- a comparing device for comparing the value of said action counter and said maximum number of actions of a respective data set, wherein said system is suitable for executing an action if said number of executed actions is lower than said maximum number of actions;
20
- increasing said action counter if an action is executed;
- an action table or list for separately storing all actions already performed with a particular data set, wherein the value of the counter is generated by summing up the number of already performed actions for a particular data set.

25
15. The computer device (10) according to claim 14, further comprising a tracking device for tracking a reaction in response to said executed action.

30

35

40

45

50

55

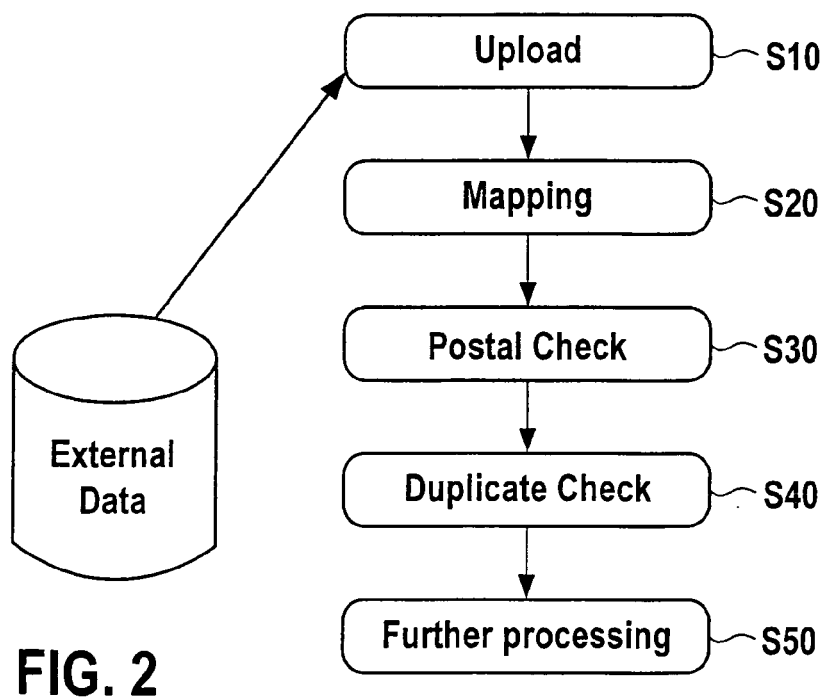
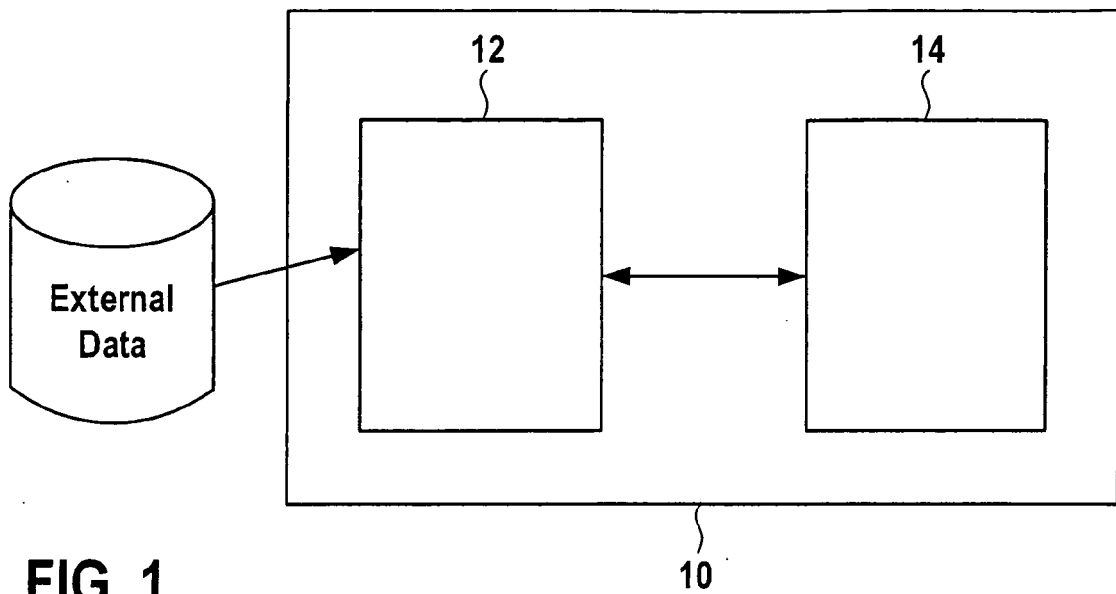


FIG. 3A

Menu: List Edit Goto Extras System Help

Toolbar: [Icons for navigation and editing]

Window Title: Create Address List: PERSON_LIST

Left Sidebar:

- Find:
 - List: [Text Box]
 - By: [Text Box]
 - ID: [Text Box]
 - Description: [Text Box]
 - Start: [Button]
- Address List: [List Box]

Main Content Area:

Address List Description: [Text Box]

Process Steps: [Text Box]

Text: [Text Box]

Data after Mapping: [Text Box]

Details: [Text Box]

Basic Data:

Origin	Provider 1	44
Type	Rented	46
Costs	750.00	EUR
Permitted No. of Uses	3	50
Permitted End-of-Use Date	31.12.2004	52
Recommended End-of-Use Date	1000	54
Gross Number		56

File Dialog:

File: C:\B2C_1.txt

Type of Field Separator: Comma

Callouts:

- 30: Address List Description
- 32: List with persons
- 34: Address List
- 36: Process Steps
- 38: Text
- 40: Data after Mapping
- 42: Details
- 44: Provider 1
- 46: Rented
- 48: 750.00
- 50: 3
- 52: 31.12.2004
- 54: 1000
- 56: Gross Number
- 60: C:\B2C_1.txt
- 62: Fields of table CRMD_MKTLIST_H and CRMD_MKTLIST_T
- 64: Fields of table CRMD_MKTLIST_PH

FIG. 3B

List Edit Goto Extras System Help 78

SAP

Create Address List: PERSON_LIST

Show/Hide Locator Favorites

Search List List ID Description Start

36 Address List PERSON_LIST List with persons

Basic Data Process Steps Text Data after Mapping Details

66 Upload File 68 Map Data 70 Postal Check Duplicate Check Business Partner Creation Target Group Creation

Continue & Process errorfree records Continue & Process errorfree records Continue & Process errorfree records Continue & Process errorfree records Continue & Process errorfree records

Formal PERSON

76 Target Group PERSON_LIST New / Existing Profile Set New Profile Set PERSON_LIST Profile Set

80 Start Immediately

Fields of tables CRMD_MKTLIST_PH and CRMD_MKTLIST_S

72 71

Address List

FIG. 3C

List Edit Goto Extras System Help

Create Address List: PERSON_LIST

Show/Hide Locator

Search Favorites

Find

By List List ID

ID

Description

Start

Address List

PERSON_LIST

List with persons

Basic Data Process Steps Text Data after Mapping Details

Language English

This list is RENTED !!!!
3 outbound contacts are allowed.

82

Ln 1 - Ln 2 of lines

Li 2, Co 33

FIG. 3D

List Edit Goto Extras System Help

Display Address List: PERSON_LIST

Show/Hide Locator

Search Favorites

Find By ID Description

Start

Address List

86

88

90

92

36

Address List Description

PERSON_LIST

List with persons

Basic Data Process Steps Text Data after Mapping Details

84

Upload File

Map Data

Postal Check

Duplicate Check

Business Partner Creation

Target Group Creation

Continue & Process erroffree records

Continue & Process erroffree records

Continue & Process erroffree records

Continue & Process erroffree records

Continue & Process erroffree records

Continue & Process erroffree records

PERSON

All Data Records

96

94

Target Group

PERSON_LIST

New / Existing Profile Set

Profile Set

PERSON_LIST

Delete temporary Data

Delete List

Start

FIG. 3E

Process Steps Log

X

6/

STOP

0

1

△

0

0

6

X

Type

Message Text

○

10 records uploaded successfully

○

Step Upload File started at 29 January 2004 09:29:40 (UTC) ended at 29 January 2004 09:29:40 (UTC)

○

10 records mapped successfully

○

Step Map Data started at 29 January 2004 09:29:41 (UTC) ended at 29 January 2004 09:29:41 (UTC)

○

5 Business Partners created

□

5 errors while creating Business Partners

○

Step Business Partner Creation started at 29 January 2004 09:29:42 (UTC) ended at 29 January 2004 09:29:46 (UTC)

✓

X

FIG. 3F

Error Analysis of Step Business Partner Creation

Start Position

1

Record Number

Record No

Field No

Msg type

Long Text

Error Msg

Data Rec

Record Number	Record No	Field No	Msg type	Long Text	Error Msg	Data Rec
1	0	0				Mr.,Jame...
Telephone/fax code for country code USV is not maintained						
7	0	0				Mr.,Franz,...
8	0	0				Mr.,Josef,...
9	0	0				Mr.,Winfri...
10	0	0				Mr.,Johan...

FIG. 3G

List Edit Goto Extras System Help

PERSON_LIST

SAP

PERSON_LIST

40

Display Address List: PERSON_LIST

Show/Hide Locator

PERSON_LIST

List with persons

Address List

Description

PERSON_LIST

List with persons

Basic Data

Process Steps

Text

Data after Mapping

Details

Start Position

1

Find

By

ID

Description

List

List ID

Start

Address List

Record No	Title	First Name	Last Na...	Second Forename or Middle Name	Date of Birth	Gender	Postal Code
1	0002	James	Edwards	T.			12345
2	0002	Leonard	McNeel	H.			77777
3	0002	Homer	Wilson	J.			98765
4	0002	Hugh	Deep				98765
5	0002	Bartholomew	Ozz				98765
6	0002	Bernd	Gelb				66763
7	0002	Franz	Schwarz				66763
8	0002	Josef	Rot				66763
9	0002	Winfried	Gelb				66763
10	0002	Johannes	Gelb				66763

FIG. 3H

ListEditGotoExtrasSystemHelp

SAP

Display Address List: PERSON_LIST

Show/Hide Locator

SearchFavoritesListIDDescriptionStart

FindByIDDescriptionAddress List

PERSON LIST42

List with persons

Basic DataProcess StepsTextData after MappingDetails

Key Figures

Gross Number	1 .000
Net Number	10
Number Postally incorrect	0
Number of Duplicates	0
No. of Business Partners (Organization)	0
Number of Business Partners (Person)	5
Number of Business Partners (Total)	5

Administrative Data

Created BySTEINMANN	Created on29.01.2004 09:28:39
Changed BySTEINMANN	Changed On29.01.2004 09:29:39

FIG. 4A

Mapping Format Edit Goto Extras System Help

SAP

Create Mapping format

Show/Hide Locator

Search Favorites

Find Mapping Format
By Mapping Format
Format

Start

Mapping Format PERSON 98
Description Mapping of person

File C:\B2C_1.TXT
Type of Field Separator Comma

Preview Row 1

Category	Target Field	Description
☒	PERS_COUNTR..	Country ISO code
☐	PERS_REGION	Region (State, Province..)
☐	PERS_ID_NUMB.	Identification Number

108 M12 112 R4 R5

Position	File Preview	Target Field
☒ 1	Mr.	PERS_TITLE_KEY
☐ 2	James	PERS_FIRSTNAME
☐ 3	T.	PERS_INITIAL

114 R3

Value Assignment Constant Assignment ABAP Code Global Definition

Possible value	Short Text	Long Text
☐ 0001		Mrs.
☒ 0002		Mr.
☐		
☐		
☐		

Assigned Values

Mr.
Mr.
Mister
MR.

106 R1 R2 122 M16

FIG. 4B

[illegible]

FIG. 4C

Mapping Format Edit Goto Extras System Help

Create Mapping format

Show/Hide Locator

Search Favorites

Find Mapping Format
By Mapping Format
Format Start

Mapping Format PERSON
Description Mapping of person

File C:\B2C_1.TXT
Type of Field Separator Comma

Position 13
File Preview
Target Field PERS_LANGUAGE

Preview Row 1

Category	Target Field	Description
☐	PERS_E_MAIL	E-MAIL Address
☐	PERS_PAGER	Pager number
☐	PERS_LANGUAGE	Business Partner: Language

Value Assignment Constant Assignment ABAP Code Global Definition

*** LS_D_DATA INPUT VARIABLE ***
 *** LS_MKTLIST_PER-PERS_LANGUAGE OUTPUT VARIABLE ***
 IF 'ls_d_data' IS INITIAL.
 * no language given => determine the language by the country
 CALL FUNCTION 'CRM_MKTLIST_COUNTRY_TO_LANG'
 EXPORTING
 iv_country_iso = gv_per_countryiso
 IMPORTING
 ev_language_iso = ls_mktlist_per-pers_language.
 ENDIF.

Ln 1 - Ln 10 of 10 lines

126

FIG. 4D

Mapping Format Edit Goto Extras System Help

Create Mapping format

Show/Hide Locator

Search Favorites

Find Mapping Format
By Mapping Format
Format Start

Mapping Format PERSON
Description Mapping of person

File C:\B2C_1.TXT
Type of Field Separator Comma

Preview Row 1

Category	Target Field	Description
☐	PERS_E_MAIL	E-MAIL Address
☐	PERS_PAGER	Pager number
☒	PERS_LANGUAGE	Business Partner: Language

Position 13
File Preview PERS_LANGUAGE
Target Field PERS_LANGUAGE

Value Assignment / Constant Assignment / ABAP Code / Global Definition

DATA: GV_PER_COUNTRYISO TYPE INTCA.

Ln 1 - Ln 2 of 2 lines

128

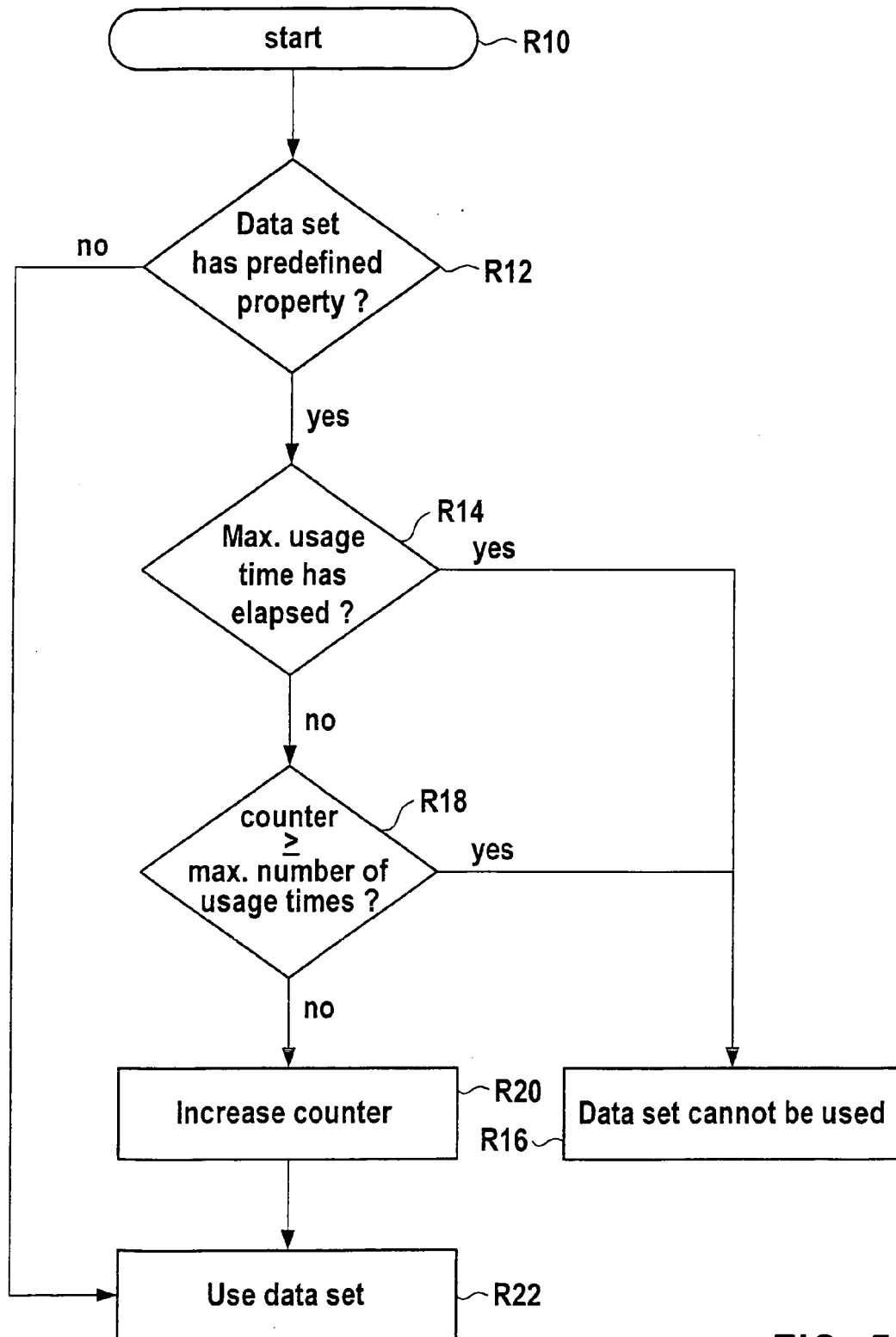


FIG. 5

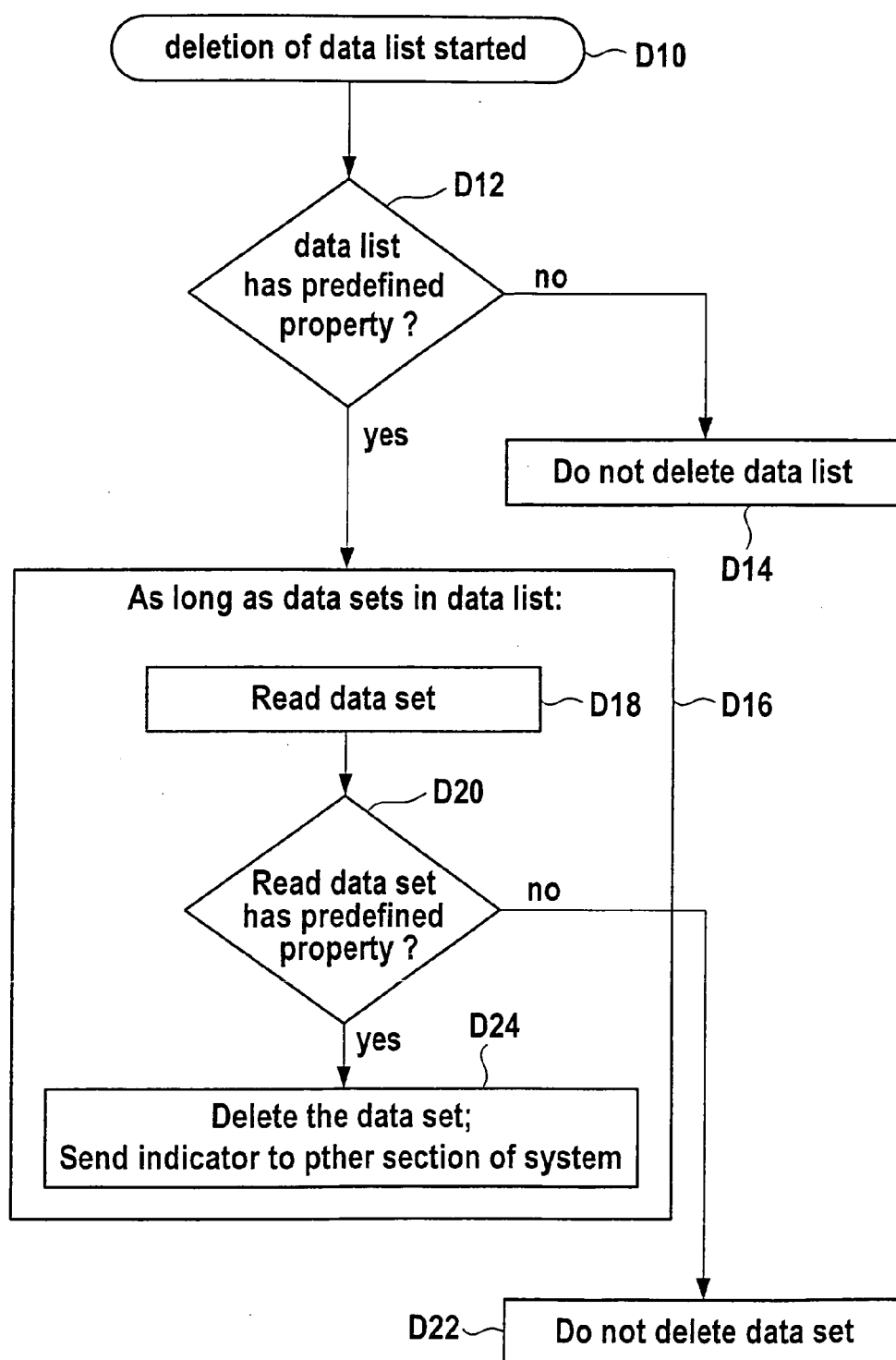


FIG. 6

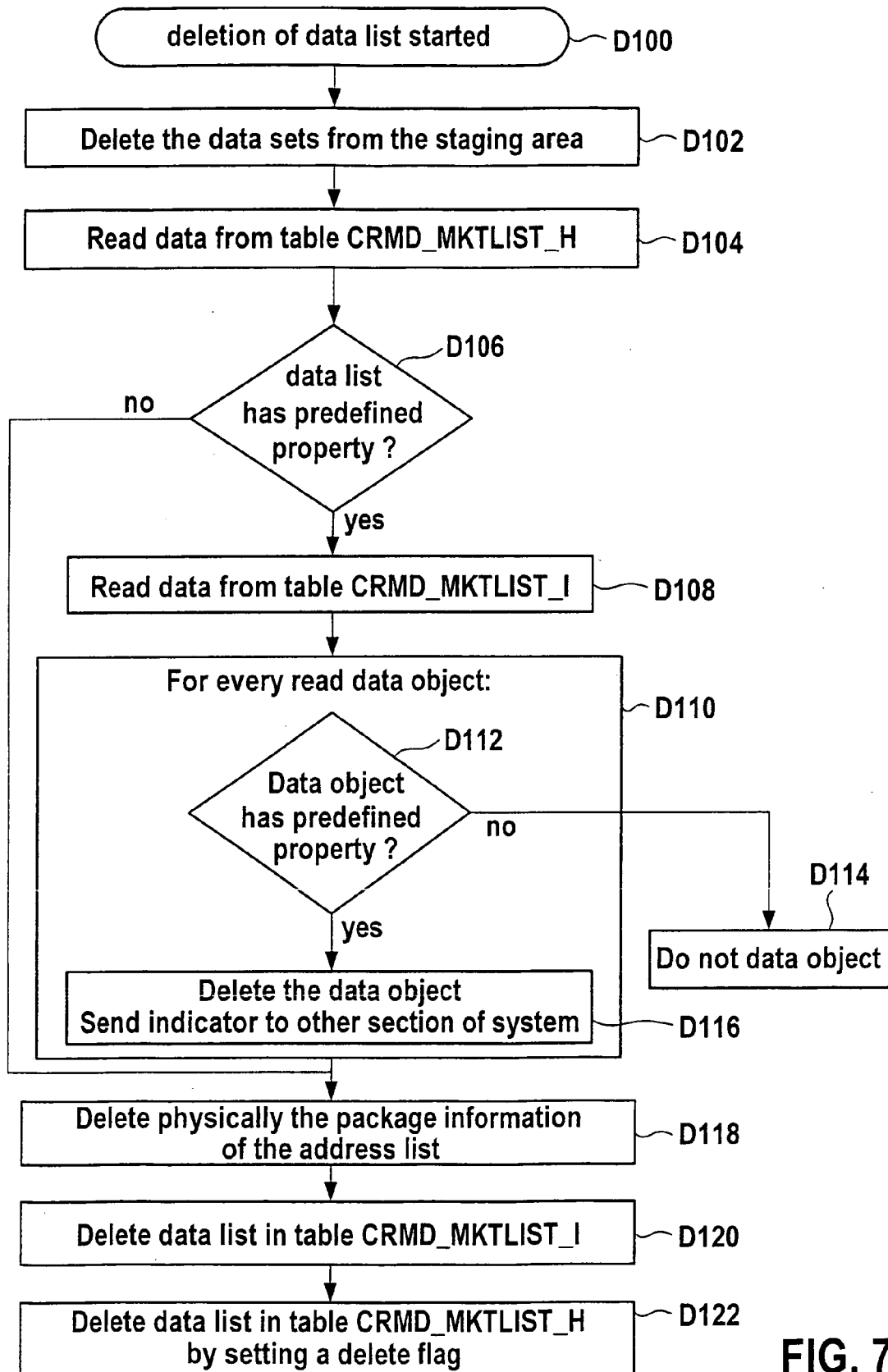


FIG. 7



PARTIAL EUROPEAN SEARCH REPORT

Application Number

under Rule 62a and/or 63 of the European Patent Convention.
This report shall be considered, for the purposes of
subsequent proceedings, as the European search report

EP 10 00 9390

DOCUMENTS CONSIDERED TO BE RELEVANT			
Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (IPC)
X	WO 03/073289 A1 (SCIENCE PARK CORP [JP]; SHOJI KOICHIRO [JP]; NOZAKI TAKASHI [JP]) 4 September 2003 (2003-09-04) * the whole document *	1-4, 13-15	INV. G06F21/00
E	& EP 1 489 516 A1 (SCIENCE PARK CORP [JP]) 22 December 2004 (2004-12-22) * abstract; figures 1,3,4,6-8 * * paragraph [0109] - paragraph [0116] *	1-4, 13-15	
A	US 2004/039741 A1 (BENSON GREG [US] ET AL) 26 February 2004 (2004-02-26) * abstract; figure 1 * * paragraphs [0002], [0020], [0025], [0052], [0091]; claim 12 *	1-4, 13-15	
A	US 5 845 281 A (BENSON GREG [SE] ET AL) 1 December 1998 (1998-12-01) * abstract; figures 1,3,17 * * column 2, line 15 - column 4, line 0 *	1-4, 13-15	
			TECHNICAL FIELDS SEARCHED (IPC)
			G06F
INCOMPLETE SEARCH			
The Search Division considers that the present application, or one or more of its claims, does/do not comply with the EPC so that only a partial search (R.62a, 63) has been carried out. Claims searched completely : Claims searched incompletely : Claims not searched : Reason for the limitation of the search: see sheet C			
Place of search		Date of completion of the search	Examiner
Munich		9 December 2010	Harms, Christoph
CATEGORY OF CITED DOCUMENTS X : particularly relevant if taken alone Y : particularly relevant if combined with another document of the same category A : technological background O : non-written disclosure P : intermediate document T : theory or principle underlying the invention E : earlier patent document, but published on, or after the filing date D : document cited in the application L : document cited for other reasons & : member of the same patent family, corresponding document			

4

EPO FORM 1503 03.82 (P04E07)



**INCOMPLETE SEARCH
SHEET C**

Application Number

EP 10 00 9390

Claim(s) completely searchable:
1-4, 13-15

Claim(s) not searched:
5-12

Reason for the limitation of the search:

A meaningful search cannot be carried out for the dependent claims 5-12 because claims 5-12 as well as their combinations with claims 1-4 are not directly and unambiguously derivable from the parent application and in particular not from page 4 lines 8-9, page 5 lines 14-15 and lines 23-26 and page 17 lines 29-32 (see applicant's letter dated 9.9.2010) of the description; Article 76(1) EPC and EPC Guidelines C-VI, 9.1.4.

**ANNEX TO THE EUROPEAN SEARCH REPORT
ON EUROPEAN PATENT APPLICATION NO.**

EP 10 00 9390

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report.
The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

09-12-2010

Patent document cited in search report		Publication date	Patent family member(s)	Publication date
WO 03073289	A1	04-09-2003	AU 2003211375 A1	09-09-2003
			CN 1620647 A	25-05-2005
			EP 1489516 A1	22-12-2004
			JP 4257909 B2	30-04-2009
			JP 4350150 B2	21-10-2009
			JP 2008171433 A	24-07-2008
			US 2010030782 A1	04-02-2010
			US 2005120359 A1	02-06-2005

US 2004039741	A1	26-02-2004	US 2005177742 A1	11-08-2005

US 5845281	A	01-12-1998	AU 4681496 A	21-08-1996
			DE 69636050 T2	23-11-2006
			EP 0807283 A2	19-11-1997
			ES 2260769 T3	01-11-2006
			JP 10513289 T	15-12-1998
			JP 2006120141 A	11-05-2006
			SE 504085 C2	04-11-1996
			SE 9500355 A	02-08-1996
			WO 9624092 A2	08-08-1996
